# Complexity of Model Checking for Modal Dependence Logic

Johannes Ebbing*, Peter Lohmann*

January 19, 2013

arXiv:1104.1034v2 [cs.LO] 27 Jan 2012

Modal dependence logic (MDL) was introduced recently by Väänänen. It enhances the basic modal language by an operator $=(\cdot)$. For propositional variables $p_1, \ldots, p_n$ the atomic formula $=(p_1, \ldots, p_{n-1}, p_n)$ intuitively states that the value of $p_n$ is determined solely by those of $p_1, \ldots, p_{n-1}$.

We show that model checking for MDL formulae over Kripke structures is *NP*-complete and further consider fragments of MDL obtained by restricting the set of allowed propositional and modal connectives. It turns out that several fragments, e.g., the one without modalities or the one without propositional connectives, remain *NP*-complete.

We also consider the restriction of MDL where the length of each single dependence atom is bounded by a number that is fixed for the whole logic. We show that the model checking problem for this bounded MDL is still *NP*-complete while for some fragments, e.g., the fragment with only $\Diamond$, the complexity drops to *P*.

We additionally extend MDL by allowing classical disjunction – introduced by Sevenster – besides dependence disjunction and show that classical disjunction is always at least as computationally bad as bounded arity dependence atoms and in some cases even worse, e.g., the fragment with nothing but the two disjunctions is *NP*-complete.

Furthermore we almost completely classifiy the computational complexity of the model checking problem for all restrictions of propositional and modal operators for both unbounded as well as bounded MDL with both classical as well as dependence disjunction.

This is the second arXiv version of this paper. It extends the first version by the investigation of the classical disjunction. A shortened variant of the first arXiv version was presented at SOFSEM 2012 [EL12].

**ACM Subject Classifiers:** F.2.2 Complexity of proof procedures; F.4.1 Modal logic; D.2.4 Model checking

**Keywords:** dependence logic, modal logic, model checking, computational complexity

*Leibniz Universität Hannover, Theoretical Computer Science, Appelstr. 4, 30167 Hannover, Germany, {ebbing,lohmann}@thi.uni-hannover.de

© Johannes Ebbing and Peter Lohmann
licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License

This work was partially supported by the NTH Focused Research School for IT Ecosystems.

# 1 Introduction

Dependence among values of variables occurs everywhere in computer science (databases, software engineering, knowledge representation, AI) but also the social sciences (human history, stock markets, etc.). In his monograph [Vää07] in 2007 Väänänen introduced functional dependence into the language of first-order logic.

Functional dependence of the value of $q$ from the values of $p_1, \dots, p_n$ means that there exists a determinating function $f$ with $q = f(p_1, \dots, p_n)$, i.e., the value of $q$ is completely determined by the values of $p_1, \dots, p_n$ alone. We denote this form of dependence (or determination) by the *dependence atom* $=(p_1, \dots, p_n, q)$. To examine dependence between situations, plays, worlds, events or observations we consider collections of these, so called *teams*. For example, a database can be interpreted as a team. In this case $=(p_1, \dots, p_n, q)$ means that in every record the value of the attribute $q$ is determined by the values of the attributes $p_1, \dots, p_n$.

In modal logic a team is a set of worlds in a Kripke structure. Here $=(p_1, \dots, p_n, q)$ means that in every world of the team the value of the atomic proposition $q$ is determined by the propositions $p_1, \dots, p_n$, i.e., there is a fixed Boolean function $f : \{0,1\}^n \to \{0,1\}$ that determines the value of $q$ from the values of $p_1, \dots, p_n$ for all worlds in the team. In first-order logic $=(x_1, \dots, x_n, y)$ means the same for a function $f : A^n \to A$ where $A$ is the universe of a first-order structure. *Dependence logic* [Vää07] is then defined by simply adding dependence atoms to usual first-order logic and *modal dependence logic* ($\mathsf{MDL}$) [Vää08, Sev09] is defined by introducing dependence atoms to modal logic.

Besides the inductive semantics (which we will use here) Väänänen also gave two equivalent game-theoretic semantics for $\mathsf{MDL}$ [Vää08]. Sevenster showed that for singleton sets of worlds there exists a translation from $\mathsf{MDL}$ to plain modal logic [Sev09]. Sevenster also showed that the satisfiability problem for $\mathsf{MDL}$ is *NEXPTIME*-complete [Sev09] and Lohmann and Vollmer continued the complexity analysis of the satisfiability problem for $\mathsf{MDL}$ by systematically restricting the set of allowed modal and propositional operators and completely classifying the complexity for all fragments of $\mathsf{MDL}$ definable in this way [LV10].

Sevenster [Sev09] also introduced classical disjunction (which is *classical* in a more set theoretic way of looking at the semantics; cf. [AV09]) into the language of $\mathsf{MDL}$. In the following we always think of the version that includes both classical disjunction (here denoted by $\varovee$) as well as dependence disjunction when we write $\mathsf{MDL}$.

The method of systematically classifying the complexity of logic related problems by restricting the set of operators allowed in formulae goes back to Lewis who used this method for the satisfiability problem of propositional logic [Lew79]. Recently it was, for example, used by Hemaspaandra et al. for the satisfiability problem of modal logic [Hem05, HSS10] and by Lohmann and Vollmer for the satisfiability problem of $\mathsf{MDL}$ [LV10]. The motivation for this approach is that by systematically examining all fragments of a logic one might find a fragment which allows for efficient algorithms but still has high enough expressivity to be useful in practice. On the other hand, this systematic approach usually leads to insights into the sources of hardness, i.e., the exact

components of the logic that make satisfiability, model checking etc. hard.

In this paper we transfer the method from satisfiability [LV10] to model checking and classify the model checking problem for almost all fragments of MDL definable by restricting the set of allowed modal ($\Box$, $\Diamond$) and propositional ($\wedge$, $\vee$, $\varovee$, $\neg$) operators to an arbitrary subset of all operators. The model checking problem asks whether a given formula is true in a given team of a given Kripke structure. For plain modal logic this problem is solvable in $P$ as shown by Clarke et al. [CES86]. A detailed complexity classification for the model checking problem over fragments of modal logic was shown by Beyersdorff et al. [BMM$^+$11] (who investigate the temporal logic CTL which contains plain modal logic as a special case).

In the case of MDL it turns out that model checking is $NP$-complete in general and that this still holds for several seemingly quite weak fragments of MDL, e.g., the one without modalities or the one where nothing except dependence atoms and $\Diamond$ is allowed (first and fourth line in Table 1). Strangely, this also holds for the case where only the both disjunctions $\vee$ and $\varovee$ are allowed and not even dependence atoms occur (third line in Table 1).

Furthermore it seems natural to not only restrict modal and propositional operators but to also impose restrictions on dependence atoms. One such restriction is to limit the arity of dependence atoms, i.e., the number $n$ of variables $p_1, \ldots, p_n$ by which $q$ has to be determined to satisfy the formula $=(p_1, \ldots, p_n, q)$, to a fixed upper bound $k \geq 0$ (the logic is then denoted by $\mathsf{MDL_k}$). For this restriction model checking remains $NP$-complete in general but for the fragment with only the $\Diamond$ operator allowed this does not hold any more (seventh line in Table 2). In this case either $\wedge$ (fourth line in Table 2) or $\vee$ (sixth line in Table 2) is needed to still get $NP$-hardness.

We classify the complexity of the model checking problem for fragments of MDL with unbounded as well as bounded arity dependence atoms. We are able to determine the tractability of each fragment except the one where formulae are built from atomic propositions and unbounded dependence atoms only by disjunction and negation (sixth line in Table 1). In each of the other cases we either show $NP$-completeness or show that the model checking problem admits an efficient (polynomial time) solution.

In Table 1 we list our complexity results for the cases with unbounded arity dependence atoms and in Table 2 for the cases with an a priori bound on the arity. In these tables a "+" means that the operator is allowed, a "-" means that the operator is forbidden and a "*" means that the operator does not affect the complexity of the problem.

## 2 Modal dependence logic

We will briefly present the syntax and semantics of MDL. For a more in-depth introduction we refer to Väänänen's definition of MDL [Vää08] and Sevenster's model-theoretic and complexity analysis [Sev09] which also contains a self-contained introduction to MDL.

**Definition 2.1.** (Syntax of MDL)
Let $AP$ be an arbitrary set of atomic propositions and $p_1, \ldots, p_n, q \in AP$. Then MDL

| **Operators** | | | | | | | **Complexity** | **Reference** |
|---|---|---|---|---|---|---|---|---|
| $\Box$ | $\Diamond$ | $\wedge$ | $\vee$ | $\neg$ | $=$ | $\varovee$ | | |
| $*$ | $*$ | $+$ | $+$ | $*$ | $+$ | $*$ | *NP*-complete | Theorem 3.2 |
| $+$ | $*$ | $*$ | $+$ | $*$ | $+$ | $*$ | *NP*-complete | Theorem 3.4 |
| $*$ | $*$ | $*$ | $+$ | $*$ | $*$ | $+$ | *NP*-complete | Theorem 5.2 |
| $*$ | $+$ | $*$ | $*$ | $*$ | $+$ | $*$ | *NP*-complete | Theorem 3.3 |
| $*$ | $+$ | $+$ | $*$ | $*$ | $*$ | $+$ | *NP*-complete | Theorem 4.7, Lemma 5.1 |
| $-$ | $-$ | $-$ | $+$ | $*$ | $+$ | $-$ | in *NP* | Proposition 3.1 |
| $*$ | $*$ | $-$ | $-$ | $*$ | $-$ | $*$ | in *P* | Theorem 5.3 |
| $*$ | $-$ | $*$ | $-$ | $*$ | $*$ | $*$ | in *P* | Theorem 3.6 |
| $*$ | $*$ | $*$ | $*$ | $*$ | $-$ | $-$ | in *P* | [CES86] |

$+$ : operator present $\quad -$ : operator absent $\quad *$ : complexity independent of operator

Table 1: Classification of complexity for fragments of $\mathsf{MDL}$-MC

is the set of all formulae built from the following rules:

$$\begin{aligned}\varphi \quad ::= \quad & \top \mid \bot \mid q \mid \neg q \mid \varphi \vee \varphi \mid \varphi \varovee \varphi \mid \varphi \wedge \varphi \mid \Box \varphi \mid \Diamond \varphi \mid \\ & =(p_1, \dots, p_n, q) \mid \neg =(p_1, \dots, p_n, q).\end{aligned}$$

Note that negation is only atomic, i.e., it is only defined for atomic propositions and dependence atoms. ⌟

We sometimes write $\Box^k$ (resp. $\Diamond^k$) for $\underbrace{\Box\Box\ldots\Box}_{k \text{ times}}$ (resp. $\underbrace{\Diamond\Diamond\ldots\Diamond}_{k \text{ times}}$). For a dependence atom $=(p_1, \dots, p_n, q)$ we define its *arity* as $n$, i.e., the arity of a dependence atom is the arity of the determinating function whose existence it asserts.

In Section 4 we will investigate the model checking problem for the following logic.

**Definition 2.2.** ($\mathsf{MDL_k}$)
$\mathsf{MDL_k}$ is the subset of $\mathsf{MDL}$ that contains all formulae which do not contain any dependence atoms whose arity is greater than $k$. ⌟

We will classify $\mathsf{MDL}$ for all fragments defined by sets of operators.

**Definition 2.3.** ($\mathsf{MDL}(M)$)
Let $M \subseteq \{\Box, \Diamond, \wedge, \vee, \varovee, \neg, \top, \bot, =\}$. By $\mathsf{MDL}(M)$ (resp. $\mathsf{MDL_k}(M)$) we denote the subset of $\mathsf{MDL}$ (resp. $\mathsf{MDL_k}$) built from atomic propositions using only operators from $M$. We sometimes write $\mathsf{MDL}(op1, op2, \dots)$ instead of $\mathsf{MDL}(\{op1, op2, \dots\})$. ⌟

$\mathsf{MDL}$ formulae are interpreted over Kripke structures.

| Operators | | | | | | | Complexity | Reference |
|---|---|---|---|---|---|---|---|---|
| $\Box$ | $\Diamond$ | $\wedge$ | $\vee$ | $\neg$ | $=$ | $\ovee$ | | |
| $*$ | $*$ | $+$ | $+$ | $*$ | $+$ | $*$ | $NP$-complete | Theorem 3.2 |
| $+$ | $*$ | $*$ | $+$ | $*$ | $+$ | $*$ | $NP$-complete | Theorem 3.4 |
| $*$ | $*$ | $*$ | $+$ | $*$ | $*$ | $+$ | $NP$-complete | Theorem 5.2 |
| $*$ | $+$ | $+$ | $*$ | $*$ | $+$ | $*$ | $NP$-complete | Theorem 4.7 |
| $*$ | $+$ | $+$ | $*$ | $*$ | $*$ | $+$ | $NP$-complete | Theorem 4.7, Lemma 5.1 |
| $*$ | $+$ | $*$ | $+$ | $*$ | $+$ | $*$ | $NP$-complete | Theorem 4.8 |
| $*$ | $*$ | $-$ | $-$ | $*$ | $*$ | $*$ | in $P$ | Theorem 5.3 |
| $*$ | $-$ | $*$ | $-$ | $*$ | $*$ | $*$ | in $P$ | Theorem 3.6 |
| $-$ | $-$ | $-$ | $*$ | $*$ | $*$ | $-$ | in $P$ | Theorem 4.5 |
| $*$ | $*$ | $*$ | $*$ | $*$ | $-$ | $-$ | in $P$ | [CES86] |

$+$ : operator present $-$ : operator absent $*$ : complexity independent of operator

Table 2: Classification of complexity for fragments of $\mathsf{MDL}_k$-MC with $k \geq 1$

**Definition 2.4.** (Kripke structure)
An *AP-Kripke structure* is a tuple $W = (S, R, \pi)$ where $S$ is an arbitrary non-empty set of *worlds*, $R \subseteq S \times S$ is the *accessibility relation* and $\pi : S \to \mathcal{P}(AP)$ is the *labeling function.* ⌟

**Definition 2.5.** (Semantics of MDL)
In contrast to common modal logics, truth of a MDL formula is not defined with respect to a single world of a Kripke structure but with respect to a set (or *team*) of worlds. Let $AP$ be a set of atomic propositions and $p, p_1, \ldots, p_n \in AP$. The *truth* of a formula $\varphi \in \mathsf{MDL}$ in a team $T \subseteq S$ of an $AP$-Kripke structure $W = (S, R, \pi)$ is denoted by

$W, T \models \varphi$ and is defined as follows:

$$
\begin{array}{lllll}
W,T & \models & \top & & \text{always holds} \\
W,T & \models & \bot & \text{iff} & T = \emptyset \\
W,T & \models & p & \text{iff} & p \in \pi(s) \text{ for all } s \in T \\
W,T & \models & \neg p & \text{iff} & p \notin \pi(s) \text{ for all } s \in T \\
W,T & \models & {=}(p_1,\dots,p_{n-1},p_n) & \text{iff} & \text{for all } s_1, s_2 \in T \text{ it holds that} \\
 & & & & \pi(s_1)\cap\{p_1,\dots,p_{n-1}\} \neq \pi(s_2)\cap\{p_1,\dots,p_{n-1}\} \\
 & & & & \text{or } \pi(s_1)\cap\{p_n\} = \pi(s_2)\cap\{p_n\} \\
W,T & \models & \neg{=}(p_1,\dots,p_{n-1},p_n) & \text{iff} & T = \emptyset \\
W,T & \models & \varphi \wedge \psi & \text{iff} & W,T \models \varphi \text{ and } W,T \models \psi \\
W,T & \models & \varphi \otimes \psi & \text{iff} & W,T \models \varphi \text{ or } W,T \models \psi \\
W,T & \models & \varphi \vee \psi & \text{iff} & \text{there are sets } T_1, T_2 \text{ with } T = T_1 \cup T_2,\ W,T_1 \models \\
 & & & & \varphi \text{ and } W,T_2 \models \psi \\
W,T & \models & \Box\varphi & \text{iff} & W, \{s' \mid \exists s \in T \text{ with } (s,s') \in R\} \models \varphi \\
W,T & \models & \Diamond\varphi & \text{iff} & \text{there is a set } T' \subseteq S \text{ such that } W,T' \models \varphi \text{ and} \\
 & & & & \text{for all } s \in T \text{ there is a } s' \in T' \text{ with } (s,s') \in R
\end{array}
$$

⌟

Note that this semantics is a conservative extension of plain modal logic semantics, i. e., it coincides with the latter for formulae which do neither contain dependence atoms nor classical disjunction. Rationales for this semantics – especially for the case of the negative dependence atom – were given by Väänänen [Vää07, p. 24].

In the remaining sections we will classify the complexity of the model checking problem for fragments of MDL and $\mathsf{MDL_k}$.

**Definition 2.6.** (MDL-MC)
Let $M \subseteq \{\Box, \Diamond, \wedge, \vee, \otimes, \neg, \top, \bot, =\}$. Then the model checking problem for $\mathsf{MDL}(M)$ (resp. $\mathsf{MDL_k}(M)$) over Kripke structures is defined as the canonical decision problem of the set

$$
\begin{array}{l}\mathsf{MDL}\text{-MC}(M) \\ (\text{resp. } \mathsf{MDL}_k\text{-MC}(M))\end{array} := \left\{ \langle W, T, \varphi \rangle \;\middle|\; \begin{array}{l} W = (S, R, \pi) \text{ a Kripke structure, } T \subseteq S, \\ \varphi \in \mathsf{MDL}(M) \text{ and } W, T \models \varphi \end{array} \right\}.
$$

⌟

We write MDL-MC for $\mathsf{MDL}\text{-MC}(\{\Box, \Diamond, \wedge, \vee, \neg, =, \otimes, \top, \bot\})$.

Before we begin with the classification we state a lemma showing that it does not matter whether we include $\top$, $\bot$ or $\neg$ in a sublogic $\mathsf{MDL}(M)$ of MDL since this does not affect the complexity of $\mathsf{MDL}\text{-MC}(M)$.

**Lemma 2.7.** *Let $M$ be an arbitrary set of* MDL *operators, i. e., $M \subseteq \{\Box, \Diamond, \wedge, \vee, \otimes, \neg, =, \bot, \top\}$. Then we have that*

$$\mathsf{MDL}\text{-MC}(M) \equiv^{\mathrm{p}}_{\mathrm{m}} \mathsf{MDL}\text{-MC}(M \setminus \{\top, \bot, \neg\}).$$ ⊣

Proof. It suffices to show $\leq^{\mathrm{p}}_{\mathrm{m}}$. So let $K = (S, R, \pi)$ a Kripke structure, $T \subseteq S$, $\varphi \in \mathsf{MDL}(M)$ and the variables of $\varphi$ among $p_1, \ldots, p_n$. Let $p'_1, \ldots, p'_n, t, f$ be fresh propositional variables. Then $K, T \models \varphi$ iff $K', T \models \varphi'$, where $K' := (S, R, \pi')$ with $\pi'$ defined by

$$\begin{aligned} \pi'(s) \cap \{t, f\} &:= \{t\}, \\ \pi'(s) \cap \{p_i, p'_i\} &:= \begin{cases} \{p_i\} & \text{if } p_i \in \pi(s), \\ \{p'_i\} & \text{if } p_i \notin \pi(s), \end{cases} \end{aligned}$$

for all $i \in \{1, \ldots, n\}$ and $s \in S$, and $\varphi \in \mathsf{MDL}(M \setminus \{\top, \bot, \neg\})$ defined by

$$\varphi' := \varphi(p'_1/\neg p_1)(p'_2/\neg p_2) \ldots (p'_n/\neg p_n)(t/\top)(f/\bot).$$

■

# 3 Unbounded arity fragments

First we will show that the most general of our problems is in *NP* and therefore all model checking problems investigated later are as well.

**Proposition 3.1.** *Let $M$ be an arbitrary set of* $\mathsf{MDL}$ *operators. Then* $\mathsf{MDL}$-MC$(M)$ *is in NP. And hence also* $\mathsf{MDL}_k$-MC$(M)$ *is in NP for every $k \geq 0$.* ⊣

Proof. The following non-deterministic top-down algorithm checks the truth of the formula $\varphi$ on the Kripke structure $W$ in the evaluation set $T$ in polynomial time.

Algorithm 1: check($W = (S, R, \pi)$,$\varphi$,$T$)

```
case φ
when φ = p
  foreach s ∈ T
    if not p ∈ π(s) then
      return false
  return true

when φ = ¬p
  foreach s ∈ T
    if p ∈ π(s) then
      return false
  return true

when φ = =(p_1,...,p_n)
  foreach (s,s') ∈ T × T
    if π(s) ∩ {p_1,...,p_{n−1}} = π(s') ∩ {p_1,...,p_{n−1}} then
          // i.e., s and s' agree on the values of the propositions p_1,...,p_{n−1}
      if (q ∈ π(s) and not q ∈ π(s')) or (not q ∈ π(s) and q ∈ π(s')) then
        return false
  return true
```

```
when φ = ¬=(p_1, ..., p_n)
  if T = ∅
    return true
  return false

when φ = ψ ∨ θ
  guess two sets of states A, B ⊆ S
  if not A ∪ B = T then
    return false
  return (check(W, A, ψ) and check(W, B, θ))

when φ = ψ ⩔ θ
  return (check(W, T, ψ) or check(W, T, θ))

when φ = ψ ∧ θ
  return (check(W, T, ψ) and check(W, T, θ))

when φ = □ψ
  T' := ∅
  foreach s' ∈ S
    foreach s ∈ T
      if (s, s') ∈ R then
        T' := T' ∪ {s'}
            // T' is the set of all successors of all states in T
  return check(W, T', ψ)

when φ = ◇ψ
  guess set of states T' ⊆ S
    foreach s ∈ T
      if there is no s' ∈ T' with (s, s') ∈ R then
        return false
            // T' contains at least one successor of every state in T
  return check(W, T', ψ)
```

Now we will see that the model checking problem is *NP*-hard and that this still holds without modalities.

**Theorem 3.2.** *Let* $M \supseteq \{\wedge, \vee, =\}$*. Then* MDL-MC$(M)$ *is NP-complete. Furthermore,* $\mathsf{MDL}_k$-MC$(M)$ *is NP-complete for every* $k \geq 0$*.* ⊣

Proof. Membership in *NP* follows from Proposition 3.1. For the hardness proof we reduce from 3SAT.

For this purpose let $\varphi = C_1 \wedge \ldots \wedge C_m$ be an arbitrary 3CNF formula with variables $x_1, \ldots, x_n$. Let $W$ be the Kripke structure $(S, R, \pi)$ over the atomic propositions

$r_1, \ldots, r_n, p_1, \ldots, p_n$ defined by

$$\begin{array}{lcl} S & := & \{s_1, \ldots, s_m\}, \\ R & := & \emptyset, \\ \pi(s_i) \cap \{r_j, p_j\} & := & \begin{cases} \{r_j, p_j\} & \text{iff } x_j \text{ occurs in } C_i \text{ positively}, \\ \{r_j\} & \text{iff } x_j \text{ occurs in } C_i \text{ negatively}, \\ \emptyset & \text{iff } x_j \text{ does not occur in } C_i. \end{cases} \end{array}$$

Let $\psi$ be the $\mathsf{MDL}(\wedge, \vee, =_0)$ formula

$$\bigvee_{j=1}^{n} r_j \wedge =(p_j)$$

and let $T := \{s_1, \ldots, s_m\}$ the evaluation set.

We will show that $\varphi \in$ 3SAT iff $W, T \models \psi$. Then it follows that 3SAT $\leq_{\mathrm{m}}^{\mathrm{p}}$ $\mathsf{MDL}_{\mathrm{o}}$-MC$(M)$ and therefore $\mathsf{MDL}_{\mathrm{o}}$-MC$(M)$ is *NP*-hard.

Now assume $\varphi \in$ 3SAT and $\theta$ an interpretation with $\theta \models \varphi$. From the valuations $\theta(x_j)$ of all $x_j$ we construct subteams $T_1, \ldots, T_n$ such that for all $j \in \{1, \ldots, n\}$ it holds that $W, T_j \models \gamma_j$ with $\gamma_j := r_j \wedge =(p_j)$. The $T_j$ are constructed as follows

$$T_j := \begin{cases} \{s_i \in S \mid \pi(s_i) \cap \{r_j, p_j\} = \{r_j, p_j\}\} & \text{iff } \theta(x_j) = 1 \\ \{s_i \in S \mid \pi(s_i) \cap \{r_j, p_j\} = \{r_j\}\} & \text{iff } \theta(x_j) = 0 \end{cases}$$

i.e., $T_j$ is the team consisting of exactly the states corresponding to clauses satisfied by $\theta(x_j)$.

Since every clause in $\varphi$ is satisfied by some valuation $\theta(x_j) = 1$ or $\theta(x_j) = 0$ we have that $T_1 \cup \ldots \cup T_n = T$ such that $W, T \models \varphi$.

On the other hand, assume that $W, T \models \psi$, therefore we have $T = T_1 \cup T_2 \cup \ldots \cup T_n$ such that for all $j \in \{1, \ldots, n\}$ it holds that $T_j \models \gamma_j$. Therefore $\pi(s_i) \cap \{p_j\}$ is constant for all elements $s_i \in T_j$. From this we can construct a valid interpretation $\theta$ for $\varphi$.

For all $j$ let $I_j := \{i \mid s_i \in T_j\}$. For every $j \in \{1, \ldots, n\}$ we consider $T_j$. If for every element $s_i \in T_j$ it holds that $\pi(s_i) \cap \{p_j\} = \{p_j\}$ then we have for all $i \in I_j$ that $x_j$ is a literal in $C_i$. In order to satisfy those $C_i$ we set $\theta(x_j) = 1$. If for every element $s_i \in T_j$ it holds that $\pi(s_i) \cap \{p_j\} = \emptyset$ then we have for every $i \in I_j$ that $\neg x_j$ is a literal in $C_i$. In order to satisfy those $C_i$ we set $\theta(x_j) = 0$.
Since for every $s_i \in T$ there is a $j$ with $s_i \in T_j$ we have an evaluation $\theta$ that satisfies every clause in $\varphi$. Therefore we have $\theta \models \varphi$. ■

Instead of not having modalities at all we can also allow nothing but the $\Diamond$ modality, i.e., we disallow propositional connectives and the $\Box$ modality, and model checking is *NP*-complete as well.

**Theorem 3.3.** *Let $M \supseteq \{\Diamond, =\}$. Then* $\mathsf{MDL}$-MC$(M)$ *is NP-complete.* $\dashv$

Proof. Membership in *NP* follows from Proposition 3.1 again.

For hardness we again reduce from 3SAT. Let $\varphi = \bigwedge_{i=1}^{m} C_i$ be an arbitrary 3CNF formula built from the variables $x_1, \dots, x_n$. Let $W$ be the Kripke structure $(S, R, \pi)$, over the atomic propositions $p_1, \dots, p_n, q$, shown in Figure 1 and formally defined by

$$\begin{aligned}
S &:= \{c_1, \dots, c_m, s_1^1, \dots, s_n^1, s_1^0, \dots, s_n^0\}, \\
R \cap \{(c_i, s_j^1), (c_i, s_j^0)\} &:= \begin{cases} \{(c_i, s_j^1)\} & \text{iff } x_j \text{ occurs in } C_i \text{ positively,} \\ \{(c_i, s_j^0)\} & \text{iff } x_j \text{ occurs in } C_i \text{ negatively,} \\ \emptyset & \text{iff } x_j \text{ does not occur in } C_i, \end{cases} \\
\pi(c_i) &:= \emptyset, \\
\pi(s_j^1) &:= \{p_j, q\}, \\
\pi(s_j^0) &:= \{p_j\}.
\end{aligned}$$

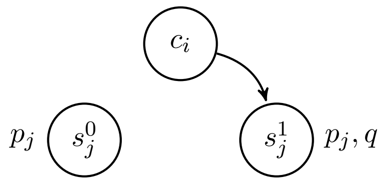


Figure 1: Kripke structure part corresponding to the 3CNF fragment $\dots \wedge C_i \wedge \dots$ with $C_i = x_j \vee \dots$.

Let $\psi$ be the $\mathsf{MDL}(\Diamond, =)$ formula

$$\Diamond = (p_1, \dots, p_n, q)$$

and let $T := \{c_1, \dots, c_m\}$.

We will show that $\varphi \in$ 3SAT iff $W, T \models \psi$. Hence, 3SAT $\leq_{\mathrm{m}}^{\mathrm{p}}$ $\mathsf{MDL}$-MC$(M)$ and $\mathsf{MDL}$-MC$(M)$ is *NP*-hard.

First suppose we have an interpretation $\theta$ that satisfies $\varphi$. From the valuations of $\theta$ we will construct a successor team $T'$ of $T$, i.e., for all $s \in T$ there is an $s' \in T'$ s.t. $(s, s') \in R$ with $W, T' \models =(p_1, \dots, p_n, q)$. $T'$ is defined by:

$$T' := \{s_j^z \mid \theta(x_j) = z, j \in \{1, \dots, n\}\}$$

Since $\theta$ satisfies every clause $C_i$ of $\varphi$ we have that for every $C_i$ there is an $x_j$ with

$$\theta(x_j) = \begin{cases} 1, & \text{iff } x_j \in C_i \\ 0, & \text{iff } \neg x_j \in C_i. \end{cases}$$

It follows that for every $s \in T$ there is an $s' \in T'$ such that $(s, s') \in R$.

By construction of $T'$ it is not possible to have both $s_j^0$ and $s_j^1$ in $T'$. Hence for all elements $s_j^0, s_{j'}^1 \in T'$ it follows that $j \neq j'$ and therefore $\pi(s_j^0) \cap \{p_1, \dots, p_n\} \neq \pi(s_{j'}^1) \cap \{p_1, \dots, p_n\}$. Thus $W, T' \models =(p_1, \dots, p_n, q)$.

On the other hand assume $W, T \models \psi$. Then there is a successor set $T'$ of $T$ s.t. for every $s \in T$ there is an $s' \in T'$ with $(s, s') \in R$ and $T' \models {=}(p_1, \dots, p_n, q)$. We construct $\theta$ as follows:

$$\theta(x_j) := \begin{cases} 1, & \text{iff } s_j^1 \in T' \\ 0, & \text{iff } s_j^0 \in T' \\ 0, & \text{iff } s_j^0, s_j^1 \notin T'. \end{cases}$$

Note that in the latter case it does not matter if 0 or 1 is chosen.

Since $W, T' \models {=}(p_1, \dots, p_n, q)$ and for every $j$ it holds that $W, \{s_j^0, s_j^1\} \not\models {=}(p_1, \dots, p_n, q)$ we have that for every $j$ at most one of $s_j^0$ or $s_j^1$ is in $T'$. It follows that $\theta$ is well-defined.

Since for every $c_i \in T$ there is an $s_j^z \in T'$ s.t. $(c_i, s_j^z) \in R$ with $\theta(x_j) = z$, we have by contruction of $W$ that $\theta$ satisfies every clause $C_i$ of $\varphi$. From this follows $\varphi \in$ 3SAT. ∎

If we disallow $\Diamond$ but allow $\Box$ instead we have to also allow $\vee$ to get *NP*-hardness.

**Theorem 3.4.** *Let $M \supseteq \{\Box, \vee, =\}$. Then* $\mathsf{MDL}$-MC$(M)$ *is NP-complete. Also,* $\mathsf{MDL}_k$-MC$(M)$ *is NP-complete for every $k \geq 0$.* ⊣

Proof. Membership in *NP* follows from Proposition 3.1 again. To prove hardness, we will once again reduce 3SAT to this problem.

Let $\varphi = \bigwedge_{i=1}^m C_i$ be an arbitrary 3CNF formula over the variables $x_1, \dots, x_n$. Let $W$ be the structure $(S, R, \pi)$, over the atomic propositions $p_1, \dots, p_n$, shown in Figures 2 to 7 and formally defined as follows:

$$\begin{aligned} S \quad := \quad & \big\{ s_i \mid i \in \{1, \dots, m\} \big\} \\ & \cup \big\{ r_k^j \mid k \in \{1, \dots, m\}, j \in \{1, \dots, n\} \big\} \\ & \cup \big\{ \overline{r}_k^j \mid k \in \{1, \dots, m\}, j \in \{1, \dots, n\} \big\} \end{aligned}$$

$$R \cap \bigcup_{j \in \{1, \dots, n\}} \{(s_i, r_i^j), (s_i, \overline{r}_i^j)\} :=$$
$$\begin{cases} \{(s_i, r_i^1)\} & \text{iff } x_1 \text{ occurs in } C_i \text{ (positively or negatively)} \quad \textit{(Fig. 2)} \\ \{(s_i, r_i^1), (s_i, \overline{r}_i^1)\} & \text{iff } x_1 \text{ does not occur in } C_i \quad \textit{(Fig. 3)} \end{cases}$$

$$R \cap \bigcup_{k \in \{1, \dots, n\}} \{(r_i^j, r_i^k), (r_i^j, \overline{r}_i^k), (\overline{r}_i^j, r_i^k), (\overline{r}_i^j, \overline{r}_i^k)\} :=$$
$$\begin{cases} \{(r_i^j, r_i^{j+1})\} & \text{iff } x_j \text{ and } x_{j+1} \text{ both occur in } C_i \quad \textit{(Fig. 4)} \\ \{(r_i^j, r_i^{j+1}), (r_i^j, \overline{r}_i^{j+1})\} & \text{iff } x_j \text{ occurs in } C_i \text{ but } x_{j+1} \text{ does not occur in } C_i \quad \textit{(Fig. 5)} \\ \{(r_i^j, r_i^{j+1}), (\overline{r}_i^j, r_i^{j+1})\} & \text{iff } x_j \text{ does not occur in } C_i \text{ but } x_{j+1} \text{ does occur in } C_i \quad \textit{(Fig. 6)} \\ \{(r_i^j, r_i^{j+1}), (\overline{r}_i^j, \overline{r}_i^{j+1})\} & \text{iff neither } x_j \text{ nor } x_{j+1} \text{ occur in } C_i \quad \textit{(Fig. 7)} \end{cases}$$

$$\begin{aligned} \pi(s_i) \quad &:= \quad \emptyset \\ \pi(r_i^j) \quad &:= \quad \begin{cases} \{p_j\} & \text{iff } x_j \text{ occurs in } C_i \text{ positively or not at all} \\ \emptyset & \text{iff } x_j \text{ occurs in } C_i \text{ negatively} \end{cases} \\ \pi(\overline{r}_i^j) \quad &:= \quad \emptyset \end{aligned}$$

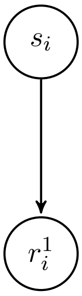


Figure 2: $x_1$ occurs in $C_i$.

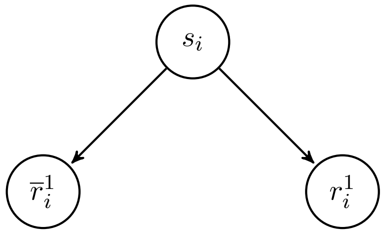


Figure 3: $x_1$ does not occur in $C_i$.

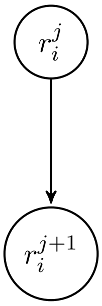


Figure 4: $x_j$ and $x_{j+1}$ occur in $C_i$.

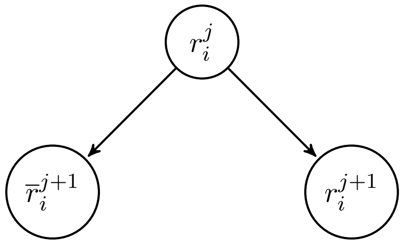


Figure 5: $x_j$ occurs in $C_i$ but $x_{j+1}$ does not occur in $C_i$.

Let $\psi$ be the $\mathsf{MDL}(\Box, \vee, =)$ formula

$$\bigvee_{j=1}^{n} \Box^j =(p_j)$$

and let $T := \{s_1, \ldots, s_m\}$.

Then, as we will show, $\varphi \in$ 3SAT iff $W, T \models \psi$ and therefore $\mathsf{MDL_o}$-MC$(\Box, \vee, =)$ is *NP*-complete. Intuitively, the direction from left to right holds because the disjunction splits the team $\{s_1, \ldots, s_m\}$ of all starting points of chains of length $n$ into $n$ subsets (one for each variable) in the following way: $s_i$ is in the subset that belongs to $x_j$ iff $x_j$

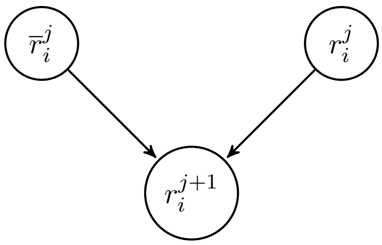


Figure 6: $x_j$ does not occur in $C_i$ but $x_{j+1}$ does occur in $C_i$.

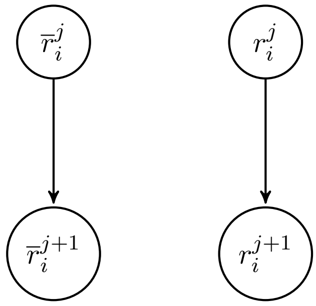


Figure 7: $x_j$ and $x_{j+1}$ do not occur in $C_i$.

satisfies the clause $C_i$ under the variable valuation that satisfies $\varphi$. Then the team that belongs to $x_j$ collectively satisfies the disjunct $\Box^j =\!(p_j)$ of $\psi$. For the reverse direction the $\overline{r}_i^j$ states are needed to ensure that a state $s_i$ can only satisfy a disjunct $\Box^j =\!(p_j)$ if there is a variable $x_j$ that occurs in clause $C_i$ (positively or negatively) and satisfies $C_i$.

More precisely, assume $\theta$ is a satisfying interpretation for $\varphi$. From $\theta$ we construct subteams $T_1, \dots, T_n$ with $T_1 \cup \ldots \cup T_n = T$ s.t. for all $j$ it holds that $T_j \models \Box^j =\!(p_j)$. $T_j$ is defined by

$$T_j := \begin{cases} \{s_i \mid \{s_i\} \models \Box^j p_j\} & \text{iff } \theta(x_j) = 1 \\ \{s_i \mid \{s_i\} \models \Box^j \neg p_j\} & \text{iff } \theta(x_j) = 0 \end{cases}$$

for all $j \in \{1, \dots, n\}$. Obviously, for all $j$ it holds that $T_j \models \Box^j =\!(p_j)$. Now we will show that for all $i \in \{1, \dots, m\}$ there is a $j \in \{1, \dots, n\}$ such that $s_i \in T_j$. For this purpose let $i \in \{1, \dots, m\}$ and suppose $C_i$ is satisfied by $\theta(x_j) = 1$ for a $j \in \{1, \dots, n\}$. Then, by definition of $W$, $\pi(r_i^j) = p_j$, hence $\{s_i\} \models \Box^j p_j$ and therefore $s_i \in T_j$. If, on the other hand, $C_i$ is satisfied by $\theta(x_j) = 0$ then we have that $\pi(r_i^j) = \emptyset$, hence $\{s_i\} \models \Box^j \neg p_j$ and again it follows that $s_i \in T_j$. Altogether we have that for all $i$ there is a $j$ such that $s_i \in T_j$. It follows that $T_1 \cup \ldots \cup T_n = T$ and therefore $W, T \models \psi$.

On the other hand assume $W, T \models \psi$. Therefore we have $T = T_1 \cup \ldots \cup T_n$ with $T_j \models \Box^j =\!(p_j)$ for all $j \in \{1, \dots, n\}$. We define a valuation $\theta$ by

$$\theta(x_j) := \begin{cases} 1 & \text{iff } T_j \models \Box^j p_j \\ 0 & \text{iff } T_j \models \Box^j \neg p_j. \end{cases}$$

Since every $s_i$ is contained in a $T_j$ we know that for all $i \in \{1, \dots, m\}$ there is a $j \in \{1, \dots, n\}$ with $\{s_i\} \models \Box^j =\!(p_j)$. From this it follows that $x_j$ occurs in $C_i$ (positively or negatively) since otherwise, by definition of $W$, both $r_i^j$ and $\overline{r}_i^j$ would be reachable from $s_i$.

It also holds that $\{s_i\} \models \Box^j p_j$ or $\{s_i\} \models \Box^j \neg p_j$. In the former case we have that $\pi(r_i^j) = p_j$, hence, by definition of $W$, $x_j$ is a literal in $C_i$. By construction of $\theta$ it follows that $C_i$ is satisfied. In the latter case it holds that $\overline{x}_j$ is a literal in $C_i$. Again, by construction of $\theta$ it follows that $C_i$ is satisfied. Hence, $\varphi \in$ 3SAT. ■

The following example demonstrates the construction from the previous proof.

**Example 3.5.** Let $\varphi$ be the 3CNF formula

$$\underbrace{(\neg x_1 \vee x_2 \vee x_3)}_{C_1} \wedge \underbrace{(x_2 \vee \neg x_3 \vee x_4)}_{C_2} \wedge \underbrace{(x_1 \vee \neg x_2)}_{C_3}.$$

The corresponding Kripke structure $W$ shown in Figure 8 has *levels* 0 to 4 where the $j$th level (corresponding to the variable $x_j$ in the formula $\varphi$) is the set of nodes reachable via exactly $j$ transitions from the set of nodes $s_1$, $s_2$ and $s_3$ (corresponding to the clauses of $\varphi$). In this example all non connected states (which do not play any role at all) are not shown.

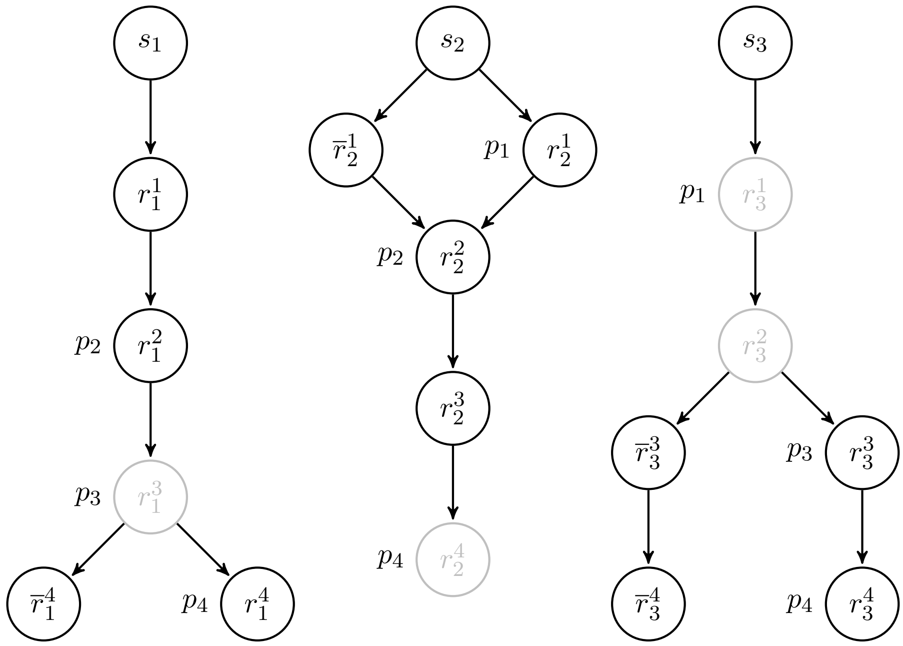


Figure 8: Kripke structure corresponding to $\varphi = (\neg x_1 \lor x_2 \lor x_3) \land (x_2 \lor \neg x_3 \lor x_4) \land (x_1 \lor \neg x_2)$

The MDL formula corresponding to $\varphi$ is

$$\psi = \underbrace{\Box{=}(p_1)}_{\gamma_1} \lor \underbrace{\Box\Box{=}(p_2)}_{\gamma_2} \lor \underbrace{\Box\Box\Box{=}(p_3)}_{\gamma_3} \lor \underbrace{\Box\Box\Box\Box{=}(p_4)}_{\gamma_4}.$$

Let $T = \{s_1, s_2, s_3\}$ with $W, T \models \psi$ and for all $j \in \{1, \ldots, 4\}$ let $T_j \subseteq T$ with $T_j \models \gamma_j$ and $T_1 \cup \ldots \cup T_4 = T$. By comparing the formulae $\gamma_j$ with the chains in the Kripke structure one can easily verify that $T_1 \subsetneq \{s_1, s_3\}$ i.e., there can at most be one of $s_1$ and $s_3$ in $T_1$ since $\pi(r_1^1) \cap p_1 \neq \pi(r_3^1) \cap \{p_1\}$ and $s_2$ cannot be in $T_1$ since its direct successors $\overline{r}_2^1, r_2^1$ do not agree on $p_1$. In this case $T_1 = \{s_1\}$ means that $C_1$ is satisfied by setting $\theta(x_1) = 0$ and the fact that $\{s_2\} \not\models \gamma_1$ corresponds to the fact that there is no way to satisfy $C_2$ via $x_1$, because $x_1$ does not occur in $C_2$. Analogously, $T_2 \subseteq \{s_1, s_2\}$ or $T_2 \subseteq \{s_3\}$, and $T_3 \subsetneq \{s_1, s_2\}$ and $T_4 \subseteq \{s_2\}$.

Now, e.g., the valuation $\theta$ where $x_1$, $x_3$ and $x_4$ evaluate to true and $x_2$ to false satisfies $\varphi$. From this valuation one can construct sets $T_1, \ldots, T_4$ with $T_1 \cup \cdots \cup T_4 = \{s_1, s_2, s_3\}$ such that $T_j \models \gamma_j$ for all $j = 1, \ldots, 4$ by defining $T_j := \{s_i \mid x_j$ satisfies clause $C_i$ under $\theta\}$ for all $j$. This leads to $T_1 = T_2 = \{s_3\}$, $T_3 = \{s_1\}$ and $T_4 = \{s_2\}$.

The gray colourings indicate which chains (resp. clauses) are satisfied on which levels (resp. by which variables). $\psi$ (resp. $\varphi$) is satisfied because there is a gray coloured state in each chain. ⌟

If we disallow both $\Diamond$ and $\vee$ the problem becomes tractable since the non-deterministic steps in the model checking algorithm are no longer needed.

**Theorem 3.6.** *Let $M \subseteq \{\Box, \wedge, \neg, =\}$. Then* $\mathsf{MDL}$-$\mathrm{MC}(M)$ *is in P.* $\dashv$

Proof. Algorithm 1 is a non-deterministic algorithm that checks the truth of an arbitrary $\mathsf{MDL}$ formula in a given structure in polynomial time. Since $M \subseteq \{\Box, \wedge, \neg, =\}$ it holds that $\Diamond, \vee \notin M$. Therefore the non-deterministic steps are never used and the algorithm is in fact deterministic in this case. ∎

Note that this deterministic polynomial time algorithm is a top-down algorithm and therefore works in a fundamentally different way than the usual deterministic polynomial time bottom-up algorithm for plain modal logic.

Now we have seen that $\mathsf{MDL}$-$\mathrm{MC}(M)$ is tractable if $\vee \notin M$ and $\Diamond \notin M$ since these two operators are the only source of non-determinism. On the other hand, $\mathsf{MDL}$-$\mathrm{MC}(M)$ is *NP*-complete if $= \in M$ and either $\Diamond \in M$ (Theorem 3.3) or $\vee, \Box \in M$ (Theorem 3.4). The remaining question is what happens if only $\vee$ (but not $\Box$) is allowed. Unfortunately this case has to remain open for now.

# 4 Bounded arity fragments

We will now show that $\mathsf{MDL}$-$\mathrm{MC}(\{\vee, \neg, =\})$ is in $P$ if we impose the following constraint on the dependence atoms in formulae given as part of problem instances: there is a constant $k \in \mathbb{N}$ such that in any input formula it holds for all dependence atoms of the form $=(p_1, \ldots, p_j, p)$ that $j \leq k$. To prove this statement we will decompose it into two smaller propositions.

First we show that even the whole $\{\vee, \neg, =\}$ fragment with unrestricted $=(\cdot)$ atoms is in $P$ as long as it is guaranteed that in every input formula at least a specific number of dependence atoms – depending on the size of the Kripke structure – occur.

We will need the following obvious lemma stating that a dependence atom is always satisfied by a team containing at least half of all the worlds.

**Lemma 4.1.** *Let $W = (S, R, \pi)$ be a Kripke structure, $\varphi := =(p_1, \ldots, p_n, q)$ ($n \geq 0$) an atomic formula and $T \subseteq S$ an arbitrary team. Then there is a set $T' \subseteq T$ such that $|T'| \geq \frac{|T|}{2}$ and $T' \models \varphi$.* $\dashv$

Proof. Let $T_0 := \{s \in T \mid q \notin \pi(s)\}$ and $T_1 := \{s \in T \mid q \in \pi(s)\}$. Then $T_0 \cup T_1 = T$ and $T_0 \cap T_1 = \emptyset$. Therefore there is an $i \in \{1, 2\}$ such that $|T_i| \geq \frac{|T|}{2}$. Let $T' := T_i$. Since $q$ is either labeled in every state of $T'$ or in no one, it holds that $W, T' \models \varphi$. ∎

We will now formalize a notion of "many dependence atoms in a formula".

**Definition 4.2.** For $\varphi \in \mathsf{MDL}$ let $\sigma(\varphi)$ be the number of positive dependence atoms in $\varphi$. Let $\ell : \mathbb{N} \to \mathbb{R}$ an arbitrary function and $\star \in \{<, \leq, >, \geq, =\}$. Then $\mathsf{MDL}$-$\mathrm{MC}_{\star\,\ell(n)}(M)$ (resp. $\mathsf{MDL}_k$-$\mathrm{MC}_{\star\ell(n)}(M)$) is the problem $\mathsf{MDL}$-$\mathrm{MC}(M)$ (resp. $\mathsf{MDL}_k$-$\mathrm{MC}(M)$) restricted to inputs $\langle W = (S, R, \pi), T, \varphi\rangle$ that satisfy the condition $\sigma(\varphi) \star \ell(|S|)$. $\lrcorner$

If we only allow $\vee$ and we are guaranteed that there are many dependence atoms in each input formula then model checking becomes trivial – even for the case of unbounded dependence atoms.

**Proposition 4.3.** *Let $M \subseteq \{\vee, \neg, =\}$. Then $\mathsf{MDL}\text{-MC}_{>\log_2(n)}(M)$ is trivial, i.e., for all Kripke structures $W = (S, R, \pi)$ and all $\varphi \in \mathsf{MDL}(M)$ such that the number of positive dependence atoms in $\varphi$ is greater than $\log_2(|S|)$ it holds for all $T \subseteq S$ that $W, T \models \varphi$.* $\dashv$

Proof. Let $W = (S, R, \pi)$, $\varphi \in \mathsf{MDL}(M)$, $T \subseteq S$ be an arbitrary instance with $\ell > \log_2(|S|)$ dependence atoms in $\varphi$. Then either $\varphi \equiv \top$ or

$$\varphi \;\equiv\; \bigvee_{i=1}^{\ell} \underbrace{=(p_{j_{i,1}}, \ldots, p_{j_{i,k_i}})}_{\psi_i} \vee \bigvee_i l_i,$$

where each $l_i$ is either a (possibly negated) atomic proposition or a negated dependence atom.

**Claim.** For all $k \in \{0, \ldots, \ell\}$ there is a set $T_k \subseteq T$ such that $W, T_k \models \bigvee_{i=1}^{k} \psi_i$ and $|T \setminus T_k| < 2^{\ell-k}$.

The main proposition follows immediately from case $k = \ell$ of this claim: From $|T \setminus T_\ell| < 2^{\ell-\ell} = 1$ follows that $T = T_\ell$ and from $W, T_\ell \models \bigvee_{i=1}^{\ell} \psi_i$ follows that $W, T \models \varphi$.

*Inductive proof of the claim.* For $k = 0$ we can choose $T_k := \emptyset$. For the inductive step let the claim be true for all $k' < k$. By Lemma 4.1 there is a set $T'_k \subseteq T \setminus T_{k-1}$ such that $W, T'_k \models \psi_k$ and $|T'_K| \geq \frac{|T \setminus T_{k-1}|}{2}$. Let $T_k := T_{k-1} \cup T'_k$. Since $W, T_{k-1} \models \bigvee_{i=1}^{k-1} \psi_i$ it follows by definition of the semantics of $\vee$ that $W, T_k \models \bigvee_{i=1}^{k} \psi_i$. Furthermore,

$$\begin{array}{rclcl} |T \setminus T_k| & = & |(T \setminus T_{k-1}) \setminus T'_k| & = & |T \setminus T_{k-1}| - |T'_k| \\ & \leq & |T \setminus T_{k-1}| - \frac{|T \setminus T_{k-1}|}{2} & = & \frac{|T \setminus T_{k-1}|}{2} \\ & < & \frac{2^{\ell-(k-1)}}{2} & = & 2^{\ell-k}. \end{array}$$ ■

Note that $\mathsf{MDL}\text{-MC}_{>\log_2(n)}(M)$ is only trivial, i.e., all instance structures satisfy all instance formulae, if we assume that only valid instances, i.e., where the number of dependence atoms is guaranteed to be large enough, are given as input. However, if we have to verify this number the problem clearly remains in $P$.

Now we consider the case in which we have very few dependence atoms (which have bounded arity) in each formula. We use the fact that there are only a few dependence atoms by searching through all possible determinating functions for the dependence atoms. Note that in this case we do not need to restrict the set of allowed $\mathsf{MDL}$ operators as we have done above.

**Proposition 4.4.** *Let $k \geq 0$. Then $\mathsf{MDL}_k\text{-MC}_{\leq\log_2(n)}$ is in $P$.* $\dashv$

Proof. From the semantics of $=$ it follows that $=(p_1, \ldots, p_k, p)$ is equivalent to

$$\exists f\, f(p_1, \ldots, p_k) \leftrightarrow p \quad := \quad \exists f\, \big((\neg f(p_1, \ldots, p_k) \vee p) \wedge (f(p_1, \ldots, p_k) \vee \neg p)\big) \qquad (1)$$

where $f(p_1, \ldots, p_k)$ and $\exists f \varphi$ – both introduced by Sevenster [Sev09, Section 4.2] – have the following semantics:

$$\begin{array}{lll}
W, T \models \exists f \varphi & \text{iff} & \text{there is a Boolean function } f^W \text{ such that} \\
 & & (W, f^W), T \models \varphi \\
(W, f^W), T \models f(p_1, \ldots, p_k) & \text{iff} & \text{for all } s \in T \text{ and for all } x_1, \ldots, x_k \in \{0,1\} \\
 & & \text{with } x_i = 1 \text{ iff } p_i \in \pi(s) \ (i = 1, \ldots, k)\text{:} \\
 & & f^W(x_1, \ldots, x_k) = 1 \\
(W, f^W), T \models \neg f(p_1, \ldots, p_k) & \text{iff} & \text{for all } s \in T \text{ and for all } x_1, \ldots, x_k \in \{0,1\} \\
 & & \text{with } x_i = 1 \text{ iff } p_i \in \pi(s) \ (i = 1, \ldots, k)\text{:} \\
 & & f^W(x_1, \ldots, x_k) = 0
\end{array}$$

Now let $W = (S, R, \pi)$, $T \subseteq S$ and $\varphi \in \mathsf{MDL}_k$ be a problem instance. First, we count the number $\ell$ of dependence atoms in $\varphi$. If $\ell > \log_2(|S|)$ we reject the input instance. Otherwise we replace every dependence atom by its translation according to 1 (each time using a new function symbol). Since the dependence atoms in $\varphi$ are at most $k$-ary we have from the transformation (1) that the introduced function variables $f_1, \ldots, f_\ell$ are also at most $k$-ary. From this it follows that the upper bound for the number of interpretations of each of them is $2^{2^k}$. For each possible tuple of interpretations $f_1^W, \ldots, f_\ell^W$ for the function variables we obtain an $\mathsf{ML}$ formula $\varphi^*$ by replacing each existential quantifier $\exists f_i$ by a Boolean formula encoding of the interpretation $f_i^W$ (for example by encoding the truth table of $f_i$ with a formula in disjunctive normal form). For each such tuple we model check $\varphi^*$. That is possible in polynomial time in $|S| + |\varphi^*|$ as shown by Clarke et al. [CES86]. Since the encoding of an arbitrary $k$-ary Boolean function has length at most $2^k$ and $k$ is constant this is a polynomial in $|S| + |\varphi|$.

Furthermore, the number of tuples over which we have to iterate is bounded by

$$\begin{array}{lll}
\left(2^{2^k}\right)^{\log_2(|S|)} & = & 2^{2^k \cdot \log_2(|S|)} \\
 & = & \left(2^{\log_2(|S|)}\right)^{2^k} \\
 & = & |S|^{2^k} \qquad \in \quad |S|^{\mathrm{O}(1)}.
\end{array}$$ ■

With Proposition 4.3 and Proposition 4.4 we have shown the following theorem.

**Theorem 4.5.** *Let $M \subseteq \{\vee, \neg, =\}$, $k \geq 0$. Then $\mathsf{MDL}_k$-MC($M$) is in P.* ⊣

Proof. Given a Kripke structure $W = (S, R, \pi)$ and a $\mathsf{MDL}_k(\vee, \neg, =)$ formula $\varphi$ the algorithm counts the number $m$ of dependence atoms in $\varphi$. If $m > \log_2(|S|)$ the input is accepted (because by Proposition 4.3 the formula is always fulfilled in this case). Otherwise the algorithm from the proof of Proposition 4.4 is used. ■

And there is another case where we can use the exhaustive determinating function search.

**Theorem 4.6.** *Let $M \subseteq \{\Box, \Diamond, \neg, =\}$. Then $\mathsf{MDL}_k$-MC($M$) is in P for every $k \geq 0$.* ⊣

Proof. Let $\varphi \in \mathsf{MDL}_k(M)$. Then there can be at most one dependence atom in $\varphi$ because $M$ only contains unary operators. Therefore we can once again use the algorithm from the proof of Proposition 4.4. ∎

In Theorem 3.3 we saw that $\mathsf{MDL}$-MC$(\Diamond, =)$ is *NP*-complete. The previous theorem includes $\mathsf{MDL}_k$-MC$(\Diamond, =) \in P$ as a special case. Hence, the question remains which are the minimal supersets $M$ of $\{\Diamond, =\}$ such that $\mathsf{MDL}_k$-MC$(M)$ is *NP*-complete.

We will now see that adding either $\wedge$ (Theorem 4.7) or $\vee$ (Theorem 4.8) is already enough to get *NP*-completeness again. But note that in the case of $\vee$ we need $k \geq 1$ while for $k = 0$ the question remains open.

**Theorem 4.7.** *Let $M \supseteq \{\Diamond, \wedge, =\}$. Then $\mathsf{MDL}_k$-MC$(M)$ is NP-complete for every $k \geq 0$.* ⊣

Proof. Membership in *NP* follows from Proposition 3.1. For hardness we once again reduce 3SAT to our problem.

For this purpose let $\varphi := \bigwedge_{i=1}^{m} C_i$ be an arbitrary 3CNF formula built from the variables $x_1, \ldots, x_n$. Let $W$ be the Kripke structure $(S, R, \pi)$ shown in Figure 9 and formally defined by

$$\begin{array}{lcl}
S & := & \{c_i \mid i \in \{1,\ldots,m\}\} \cup \{s_{j,j'}, \overline{s}_{j,j'} \mid j, j' \in \{1,\ldots,n\}\} \\
 & & \cup\ \{t_j, \overline{t}_j \mid j \in \{1,\ldots,n\}\} \\
R & := & \{(c_i, s_{1,j}) \mid x_j \in C_i\} \cup \{(c_i, \overline{s}_{1,j}) \mid \overline{x}_j \in C_i\} \\
 & & \cup\ \{(s_{k,j}, s_{k+1,j}) \mid j \in \{1,\ldots,n\}, k \in \{1,\ldots,n-1\}\} \\
 & & \cup\ \{(\overline{s}_{k,j}, \overline{s}_{k+1,j}) \mid j \in \{1,\ldots,n\}, k \in \{1,\ldots,n-1\}\} \\
 & & \cup\ \{(s_{k,j}, t_j), (\overline{s}_{k,j}, \overline{t}_j) \mid j \in \{1,\ldots,n\}, k \in \{1,\ldots,n\}\} \\
 & & \cup\ \{(s_{k,j}, \overline{t}_j), (\overline{s}_{k,j}, t_j) \mid j \in \{1,\ldots,n\}, k \in \{1,\ldots,n\}, j \neq k\} \\
\pi(c_i) & := & \emptyset \\
\pi(s_{j,j'}) & := & \emptyset \\
\pi(\overline{s}_{j,j'}) & := & \emptyset \\
\pi(t_j) & := & \{r_j, p_j\} \\
\pi(\overline{t}_j) & := & \{r_j\}.
\end{array}$$

And let $\psi$ be the $\mathsf{MDL}(\Diamond, \wedge, =)$ formula

$$\begin{array}{cl}
 & \Diamond \left( \bigwedge_{j=1}^{n} \Diamond^j (r_j \wedge {=}(p_j)) \right) \\
= & \Diamond \big( \Diamond(r_1 \wedge {=}(p_1)) \wedge \Diamond\Diamond(r_2 \wedge {=}(p_2)) \wedge \ldots \wedge \Diamond^n(r_n \wedge {=}(p_n)) \big).
\end{array}$$

We again show that $\varphi \in$ 3SAT iff $W, \{c_1, \ldots, c_m\} \models \psi$. First assume that $\varphi \in$ 3SAT and that $\theta$ is a satisfying valuation for the variables in $\varphi$. Now let

$$s_j := \begin{cases} s_{1,j} & \text{if } x_j \text{ evaluates to true under } \theta \\ \overline{s}_{1,j} & \text{if } x_j \text{ evaluates to false under } \theta \end{cases}$$

for all $j = 1, \ldots, n$. Then it holds that $W, \{s_1, \ldots, s_n\} \models \bigwedge_{j=1}^{n} \Diamond^j (r_j \wedge {=}(p_j))$.

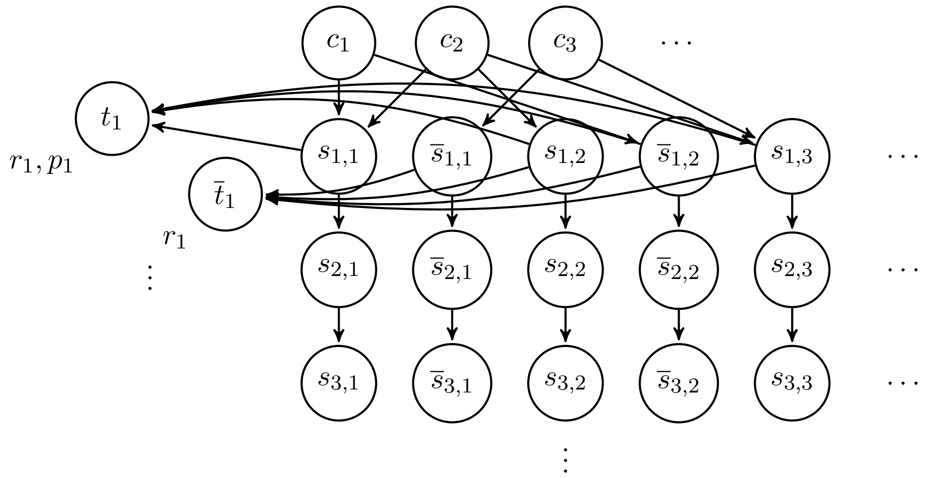


Figure 9: Kripke structure construction in the proof of Theorem 4.7
The underlying 3CNF formula contains the clauses $C_1 = x_1 \vee \neg x_2$, $C_2 = x_1 \vee x_2 \vee x_3$ and $C_3 = \neg x_1 \vee x_3$

Furthermore, since $\theta$ satisfies $\varphi$ it holds for all $i = 1, \ldots, m$ that there is a $j_i \in \{1, \ldots, n\}$ such that $(c_i, s_{j_i}) \in R$. Hence, $W, \{c_1, \ldots, c_m\} \models \Diamond \left( \bigwedge_{j=1}^{n} \Diamond^j (r_j \wedge {=}(p_j)) \right)$.

For the reverse direction assume that $W, \{c_1, \ldots, c_m\} \models \psi$. Now let $T \subseteq \{s_{1,1}, \overline{s}_{1,1}, s_{1,2}, \ldots, \overline{s}_{1,n}\}$ such that $T \models \bigwedge_{j=1}^{n} \Diamond^j (r_j \wedge {=}(p_j))$ and for all $i = 1, \ldots, m$ there is a $s \in T$ with $(c_i, s) \in R$.

Since $T \models \Diamond^j (r_j \wedge {=}(p_j))$ there is no $j \in \{1, \ldots, n\}$ with $s_{1,j} \in T$ and also $\overline{s}_{1,j} \in T$. Now let $\theta$ be the valuation of $x_1, \ldots, x_n$ defined by

$$\theta(x_j) := \begin{cases} 1 & \text{if } s_{1,j} \in T \\ 0 & \text{else.} \end{cases}$$

Since for each $i = 1, \ldots, m$ there is a $j \in \{1, \ldots, n\}$ such that either $(c_i, s_{1,j}) \in R$ and $s_{1,j} \in T$ or $(c_i, \overline{s}_{1,j}) \in R$ and $\overline{s}_{1,j} \in T$ it follows that for each clause $C_i$ of $\varphi$ there is a $j \in \{1, \ldots, n\}$ such that $x_j$ satisfies $C_i$ under $\theta$. ■

**Theorem 4.8.** *Let $M \supseteq \{\Diamond, \vee, =\}$. Then $\mathsf{MDL}_k$-MC$(M)$ is NP-complete for every $k \geq 1$.* ⊣

Proof. As above membership in *NP* follows from Proposition 3.1 and for hardness we reduce 3SAT to our problem.

For this purpose let $\varphi := \bigwedge_{i=1}^{m} C_i$ be an arbitrary 3CNF formula built from the variables $p_1, \ldots, p_n$. Let $W$ be the Kripke structure $(S, R, \pi)$ shown in Figure 10 and

formally defined by

$$
\begin{array}{lll}
S & := & \{c_{i,j} \mid i \in \{1,\dots,m\}, j \in \{1,\dots,n\}\} \cup \{x_{j,j'} \mid j, j' \in \{1,\dots,n\}, j' \leq j\} \\
R & := & \{(c_{i,j}, c_{i,j+1}) \mid i \in \{1,\dots,m\}, j \in \{1,\dots,n-1\}\} \\
 & & \cup \{(x_{j,j'}, x_{j,j'+1}) \mid j \in \{1,\dots,n\}, j' \in \{1,\dots,j-1\}\} \\
\pi(x_{j,j'}) & := & \begin{cases} \{q, p_j\} & \text{iff } j' = j \\ \{q\} & \text{iff } j' < j \end{cases} \\
\pi(c_{i,j}) & := & \begin{cases} \{q\} & \text{iff } p_j, \neg p_j \notin C_i \\ \{p_j\} & \text{iff } p_j \in C_i \\ \emptyset & \text{iff } \neg p_j \in C_i \end{cases}
\end{array}
$$

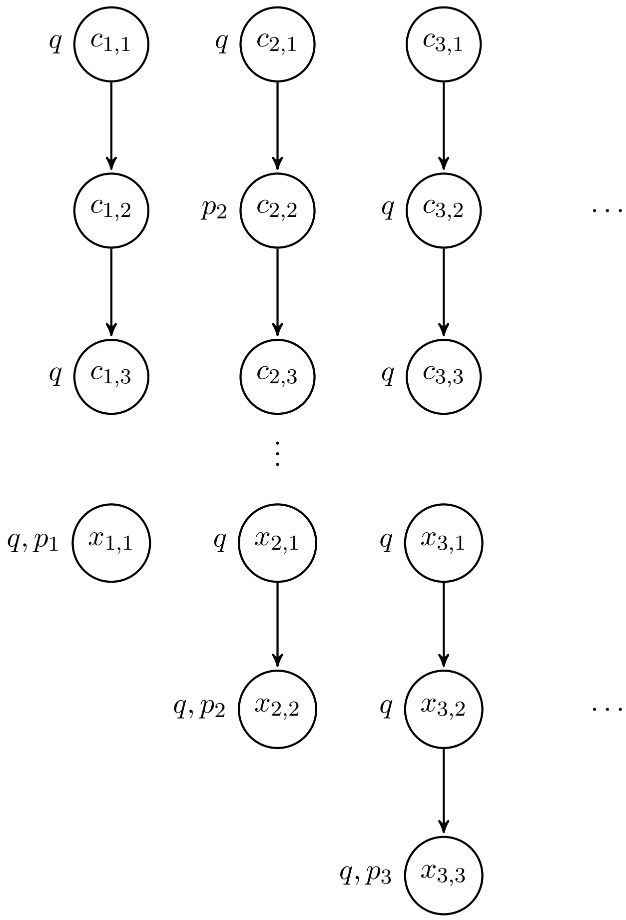


Figure 10: Kripke structure construction in the proof of Theorem 4.8
The underlying 3CNF formula contains the clauses $C_1 = \neg p_2$, $C_2 = p_2 \vee \neg p_3$ and $C_3 = \neg p_1$

Let $\psi$ be the $\mathsf{MDL}$ formula

$$\begin{aligned} & \bigvee_{j=1}^{n} \Diamond^{j-1} {=}(q, p_j) \\ \equiv \quad & {=}(q, p_1) \vee \Diamond {=}(q, p_2) \vee \Diamond \Diamond {=}(q, p_3) \vee \cdots \vee \Diamond^{n-1} {=}(q, p_n). \end{aligned}$$

Once again we show that $\varphi \in$ 3SAT iff $W, \{c_{1,1}, \ldots, c_{m,1}, x_{1,1}, x_{2,1}, \ldots, x_{n,1}\} \models \psi$. First assume that $\varphi \in$ 3SAT and that $\theta$ is a satisfying valuation for the variables in $\varphi$. Now let $P_j := \{c_{i,1} \mid C_i$ is satisfied by $p_j$ under $\theta\}$ for all $j = 1, \ldots, n$. Then it follows that $\bigcup\limits_{j=1}^{n} P_j = \{c_{1,1}, \ldots, c_{m,1}\}$ and that

$$W, P_j \models \Diamond^{j-1}(\neg q \wedge {=}(p_j))$$

for all $j = 1, \ldots, n$. Additionally, it holds that $W, \{x_{j,1}\} \models \Diamond^{j-1}(q \wedge {=}(p_j))$ $(j = 1, \ldots, n)$.

Together it follows that $W, P_j \cup \{x_{j,1}\} \models \Diamond^{j-1} {=}(q, p_j)$ for all $j = 1, \ldots, n$. This implies

$$W, \bigcup_{j=1}^{n} (P_j \cup \{x_{j,1}\}) \models \bigvee_{j=1}^{n} \Diamond^{j-1} {=}(q, p_j)$$

which is equivalent to

$$W, \{c_{1,1}, \ldots, c_{m,1}, x_{1,1}, x_{2,1}, \ldots, x_{n,1}\} \models \psi.$$

For the reverse direction assume that $W, T \models \psi$ with $T := \{c_{1,1}, \ldots, c_{m,1}, x_{1,1}, x_{2,1}, \ldots, x_{n,1}\}$. Let $T_1, \ldots, T_n$ be subsets of $T$ with $T_1 \cup \cdots \cup T_n = T$ such that for all $j \in \{1, \ldots, n\}$ it holds that $T_j \models \Diamond^{j-1} {=}(q, p_j)$. Then it follows that $x_{1,1} \in T_1$ since the chain starting in $x_{1,1}$ consists of only one state. From $\pi(x_{1,1}) = \{q, p_1\}$ and $\pi(x_{2,1}) = \{q\}$ it follows that $x_{2,1} \notin T_1$ and hence (again because of the length of the chain) $x_{2,1} \in T_2$. Inductively, it follows that $x_{j,1} \in T_j$ for all $j = 1, \ldots, n$.

Now, it follows from $x_{j,1} \in T_j$ that for all $i \in \{1, \ldots, m\}$ with $c_{i,1} \in T_j$: $q \notin \pi(c_{i,j})$ (because $q, p_j \in \pi(x_{j,j}$, $p_j \notin \pi(x_{i,j}))$. Since $T_j \models \Diamond^{j-1} {=}(q, p_j)$, it then holds that $T_j \setminus \{x_{j,1}\} \models \Diamond^{j-1}(\neg q \wedge {=}(p_j))$.

Now let $\theta$ be the valuation of $p_1, \ldots, p_n$ defined by

$$\theta(p_j) := \begin{cases} 1 & \text{if } T_j \setminus \{x_{j,1}\} \models \Diamond^{j-1}(\neg q \wedge p_j) \\ 0 & \text{if } T_j \setminus \{x_{j,1}\} \models \Diamond^{j-1}(\neg q \wedge \neg p_j) \end{cases}.$$

Since for each $i = 1, \ldots, m$ there is a $j \in \{1, \ldots, n\}$ such that $c_{i,1} \in T_j$ it follows that for each clause $C_i$ of $\varphi$ there is a $j \in \{1, \ldots, n\}$ such that $p_j$ satisfies $C_i$ under $\theta$. ∎

# 5 Classical disjunction

First we show that classical disjunction can substitute zero-ary dependence atoms.

**Lemma 5.1.** *Let* $=, \ovee \notin M$. *Then*

$$\mathsf{MDL_o}\text{-MC}(M \cup \{=\}) \leq^{\mathrm{p}}_{\mathrm{m}} \mathsf{MDL}\text{-MC}(M \cup \{\ovee\}). \qquad \dashv$$

Proof. Follows immediately from the equivalence of $=(p)$ and $p \ovee \neg p$ together with Lemma 2.7. ∎

The following surprising result shows that both kinds of disjunctions together are already enough to get *NP*-completeness.

**Theorem 5.2.** *Let* $\{\vee, \ovee\} \subseteq M$. *Then* $\mathsf{MDL}_k$-MC$(M)$ *is NP-complete for every* $k \geq 0$. $\dashv$

Proof. As above membership in *NP* follows from Proposition 3.1 and for hardness we reduce 3SAT to our problem – using a construction that bears some similarities with the one used in the proof of Theorem 4.8.

For this purpose let $\varphi := \bigwedge_{i=1}^{m} C_i$ be an arbitrary 3CNF formula built from the variables $p_1, \ldots, p_n$. Let $W$ be the Kripke structure $(S, R, \pi)$ shown in Figure 11 and formally defined by

$$\begin{array}{lcl} S & := & \{c_i \mid i \in \{1, \ldots, m\}\} \\ R & := & \emptyset \\ \pi(c_i) & := & \{p_j \mid p_j \in C_i\} \cup \{q_j \mid \neg p_j \in C_i\}. \end{array}$$

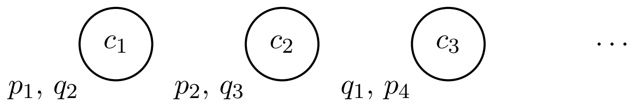


Figure 11: Kripke structure construction in the proof of Theorem 5.2
The underlying 3CNF formula contains the clauses $C_1 = p_1 \vee \neg p_2$, $C_2 = p_2 \vee \neg p_3$ and $C_3 = \neg p_1 \vee p_4$

Let $\psi$ be the $\mathsf{MDL}$ formula

$$\bigvee_{j=1}^{n} (p_j \ovee q_j).$$

Once again we show that $\varphi \in$ 3SAT iff $W, \{c_1, \ldots, c_m\} \models \psi$. First assume that $\varphi \in$ 3SAT and that $\theta$ is a satisfying valuation for $\varphi$. Now let

$$P_j := \{c_i \mid C_i \text{ is satisfied by } p_j \text{ under } \theta\}$$

for all $j = 1, \ldots, n$. Then it follows that $\bigcup_{j=1}^{n} P_j = \{c_1, \ldots, c_m\}$ and that

$$W, P_j \models p_j \ovee q_j$$

for all $j = 1, \ldots, n$. Together it follows that

$$W, \{c_1, \ldots, c_m\} \models \bigvee_{j=1}^{n} (p_j \varovee q_j).$$

For the reverse direction assume that $W, T \models \psi$ with $T := \{c_1, \ldots, c_m\}$. Let $T_1, \ldots, T_n$ be subsets of $T$ with $T_1 \cup \cdots \cup T_n = T$ such that for all $j \in \{1, \ldots, n\}$ it holds that $T_j \models p_j \varovee q_j$. Now let $\theta$ be the valuation of $p_1, \ldots, p_n$ defined by

$$\theta(p_j) := \begin{cases} 1 & \text{if } T_j \models p_j \\ 0 & \text{if } T_j \models q_j \end{cases}.$$

Since for each $i = 1, \ldots, m$ there is a $j \in \{1, \ldots, n\}$ such that $c_i \in T_j$ it follows that for each clause $C_i$ of $\varphi$ there is a $j \in \{1, \ldots, n\}$ such that $p_j$ satisfies $C_i$ under $\theta$. ∎

Now we show that Theorem 4.6 still holds if we additionally allow classical disjunction.

**Theorem 5.3.** *Let $M \subseteq \{\Box, \Diamond, \varovee, \neg, =\}$. Then $\mathsf{MDL}_k$-MC$(M)$ is in P for every $k \geq 0$.* ⊣

Proof. Let $\varphi \in \mathsf{MDL}(M)$. Because of the distributivity of $\varovee$ with all other $\mathsf{MDL}$ operators there is a formula $\psi$ equivalent to $\varphi$ which is of the form

$$\bigovee_{i=1}^{|\varphi|} \psi_i$$

with $\psi_i \in \mathsf{MDL}(M \setminus \{\varovee\})$ for all $i \in \{1, \ldots, |\varphi|\}$. Note that there are only linearly many formulas $\psi_i$ because $\varphi$ does not contain any binary operators aside from $\varovee$. Further note that $\psi$ can be easily computed from $\varphi$ in polynomial time.

Now it is easy to check for a given structure $W$ and team $T$ whether $W, T \models \psi$ by simply checking whether $W, T \models \psi_i$ (which can be done in polynomial time by Theorem 4.6) consecutively for all $i \in \{1, \ldots, |\varphi|\}$. ∎

# 6 Conclusion

In this paper we showed that MDL-MC is *NP*-complete (Theorem 3.2). Furthermore we have systematically analyzed the complexity of model checking for fragments of MDL defined by restricting the set of modal and propositional operators. It turned out that there are several fragments which stay *NP*-complete, e.g., the fragment obtained by restricting the set of operators to only $\Box, \vee$ and $=$ (Theorem 3.4) or only $\Diamond$ and $=$ (Theorem 3.3). Intuitively, in the former case the *NP*-hardness arises from existentially guessing partitions of teams while evaluating disjunctions and in the latter from existentially guessing successor teams while evaluating $\Diamond$ operators. Consequently, if we allow all operators except $\Diamond$ and $\vee$ the complexity drops to $P$ (Theorem 3.6).

For the fragment only containing $\vee$ and $=$ on the other hand we were not able to determine whether its model checking problem is tractable. Our inability to prove either *NP*-hardness or containment in *P* led us to restrict the arity of the dependence atoms. For the aforementioned fragment the complexity drops to *P* in the case of bounded arity (Theorem 4.8). Furthermore, some of the cases which are known to be *NP*-complete for the unbounded case drop to *P* in the bounded arity case as well (Theorem 4.6) while others remain *NP*-complete but require a new proof technique (Theorems 4.7 and 4.8). Most noteworthy in this context are probably the results concerning the $\Diamond$ operator. With unbounded dependence atoms this operator alone suffices to get *NP*-completeness whereas with bounded dependence atoms it needs the additional expressiveness of either $\wedge$ or $\vee$ to get *NP*-hardness.

Considering the classical disjunction operator $\otimes$, we showed that the complexity of $\mathsf{MDL}_k$-$\mathrm{MC}(M \cup \{=\})$ is never higher than the complexity of $\mathsf{MDL}_k$-$\mathrm{MC}(M \cup \{\otimes\})$, i.e., $\otimes$ is at least as bad as $=(\cdot)$ with respect to the complexity of model-checking (in contrast to the complexity of satisfiability; cf. [LV10]). And in the case where only $\vee$ is allowed we even have a higher complexity with $\otimes$ (Theorem 5.2) than with $=$ (Theorem 4.5). The case of $\mathsf{MDL}$-$\mathrm{MC}(\vee, \otimes)$ is also our probably most surprising result since the non-determinism of the $\vee$ operator turned out to be powerful enough to lead to *NP*-completeness although neither conjunction nor dependence atoms (which also, in a sense, contain some special kind of conjunction) are allowed.

Interestingly, in none of our reductions to show *NP*-hardness the $\mathsf{MDL}$ formula depends on anything else but the number of propositional variables of the input 3CNF formula. The structure of the input formula is always encoded by the Kripke structure alone. So it seems that even for a fixed formula the model checking problem could still be hard. This, however, cannot be the case since, by Theorem 4.4, model checking for a fixed formula is always in *P*.

Another open question, apart from the unclassified unbounded arity case, is related to a case with bounded arity dependence atoms. In Theorem 4.8 it was only possible to prove *NP*-hardness for arity at least one and it is not known what happens in the case where the arity is zero. Additionally, it might be interesting to determine the exact complexity for the cases which are in *P* since we have not shown any lower bounds in these cases so far.

# References


[AV09] Samson Abramsky and Jouko Väänänen. From IF to BI. *Synthese*, 167(2):207–230, 2009.

[BMM+11] Olaf Beyersdorff, Arne Meier, Martin Mundhenk, Thomas Schneider, Michael Thomas, and Heribert Vollmer. Model checking CTL is almost always inherently sequential. *Logical Methods in Computer Science*, 2011.

[CES86] E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic verification of

finite-state concurrent systems using temporal logic specifications. *ACM Trans. Program. Lang. Syst.*, 8(2):244–263, 1986.

[EL12] Johannes Ebbing and Peter Lohmann. Complexity of model checking for modal dependence logic. In *Proceedings SOFSEM 2012: Theory and Practice of Computer Science*, volume 7147 of *Lecture Notes in Computer Science*, pages 226–237. Springer Berlin / Heidelberg, 2012.

[Hem05] Edith Hemaspaandra. The complexity of poor man's logic. *CoRR*, cs.LO/9911014v2, 2005.

[HSS10] Edith Hemaspaandra, Henning Schnoor, and Ilka Schnoor. Generalized modal satisfiability. *J. Comput. Syst. Sci.*, 76(7):561–578, 2010.

[Lew79] Harry Lewis. Satisfiability problems for propositional calculi. *Mathematical Systems Theory*, 13:45–53, 1979.

[LV10] Peter Lohmann and Heribert Vollmer. Complexity results for modal dependence logic. In *Proceedings 19th Conference on Computer Science Logic*, volume 6247 of *Lecture Notes in Computer Science*, pages 411–425. Springer Berlin / Heidelberg, 2010.

[Sev09] Merlijn Sevenster. Model-theoretic and computational properties of modal dependence logic. *Journal of Logic and Computation*, 19(6):1157–1173, 2009.

[Vää07] Jouko Väänänen. *Dependence logic: A new approach to independence friendly logic*. Number 70 in London Mathematical Society student texts. Cambridge University Press, 2007.

[Vää08] Jouko Väänänen. Modal dependence logic. In Krzysztof R. Apt and Robert van Rooij, editors, *New Perspectives on Games and Interaction*, volume 4 of *Texts in Logic and Games*, pages 237–254. Amsterdam University Press, 2008.